

	Tag	Attributes	Text	Description
105 ~	<cli>	-	-	Contains zero or more modes
110 ~	<mode>	name	-	Contains zero or more command tags. Attribute is required.
115 ~	<command>	-	-	Contains a required keyword tag.
120 ~	<keyword>	text	-	Contains zero or more keyword tags, with an optional action or help tag. Attribute is required.
125 ~	<help>	-	Yes	May be embedded in keyword or arg elements.
130 ~	<action>	object method	-	Contains zero or more arg tags. The attributes are required.
135 ~	<arg>	-	-	Contains a single instance of type tag, may contain a help tag.
140 ~	<type>	Type	-	Empty tag. Attribute is required, must be a valid java type.

**FIGURE 1**

200 ~ Command> show clock  
14 10 19 109 UTC Thu May 6 1993

205 ~ Command> show clock detail  
14 10 19 109 UTC Thu May 6 1993  
Time source is user configuration

210 {  
 < xml version="1.0" encoding="US-ASCII">  
 <!DOCTYPE cli SYSTEM "cli.dtd">  
 <cli>  
 <mode name="global">  
 ....  
 <command>  
 <keyword text="show">  
 <help>Show running system information</help>  
 .... <!--lots of other show keywords... -->  
 <keyword text="clock">  
 <?help>display the system clock</help> ~ 212  
 <action object="Clock" method="show">  
 </action>  
 <keyword text="detail">  
 <help>Display detailed information </help>  
 action object="Clock" method="showDetail">  
 </action>  
 </keyword> ~ 215  
 </keyword>  
 </command>  
 </mode>  
 </cli>

FIGURE 2

- 321 ~ Command > clock set  
on or as Current Time
- 322 ~ Command > clock set 00 00 00  
<1-31> Day of the month  
MONTH Month of the year
- 323 ~ Command > clock set 00 00 00 25 Sep  
<1993-2035> Year
- 324 ~ Command > clock set 00:00 00 25 Sep 2000  
Command >

325 ~ {

```

<command>
  <keyword text="clock"> ~ 327
    <help>Manage the System clock</help>
    <keyword text="set"> ~ 328
      <help> Set the time and date</help>
      <action object="Clock" method="set"> ~ 329
        <arg> <!--time argument -->
          <type type="java.lang.String"></type>
          <help>hh:mm:ss Current Time </help> ~ 330
        </arg>
        <arg> <!--day of the month argument -->
          <type type="java.lang.Integer"></type>
          <help>&lt;1-31&gt; Day of the month</help> ~ 335
        </arg>
        <arg> <!--day of the month argument -->
          <type type="java.lang.Integer"></type>
          <help>MONTH Month of the year </help> ~ 340
        </arg>
        <arg> <!--year argument -->
          <type type="java.lang.Integer"></type>
          <help>&lt;1993-2035&gt; Year</help> ~ 345
        </arg>
      </action>
    </keyword>
  </keyword>
</command>

```

FIGURE 3

```

<?xml version='1.0' encoding='us-ascii'?>
<!--DTD for the cli description file -->

    <!--The cli tag is the topmost element. Contains zero or more modes. -->
<!ELEMENT cli (mode*)>

    <!--The mode tag contains zero or more command tags. The mode must
    have a name, for example "global", "exec", "config" -->
<!ELEMENT mode (command*)>
<!ATTLIST mode
    name CDATA #REQUIRED
>

    <!--The command tag contains a single keyword -->
<!ELEMENT command (keyword)>

    <!--The keyword tag contains zero or more subkeywords, with an optional
    single instance of action or help. The single attribute is the keyword test
    which must be present -->
<!ELEMENT keyword ((keyword)*, action?, help?)>
<!ATTLIST keyword
    test CDATA #REQUIRED
>

    <!-- The action tag contains zero or more argument tags. The "object"
    and "method" attributes must be present and defined"
<!ELEMENT action (arg*)>
<!ATTLIST action
    object CDATA #REQUIRED
    method CDATA #REQUIRED
>

    <!-- The arg tag contains a single instance of a type tag and an
    optional help tag -->
<!ELEMENT arg (type, help?)>

    <!-- The type tag is an empty tag with a required attribute "type". The
    value of the attribute should be a Java class name -->
<!ELEMENT type EMPTY>
<!ATTLIST arg
    type CDATA #REQUIRED
>

    <!-- The help tag encloses some text -->
<!ELEMENT help (#PCDATA)>

```

**FIGURE 4**

FIG. 5

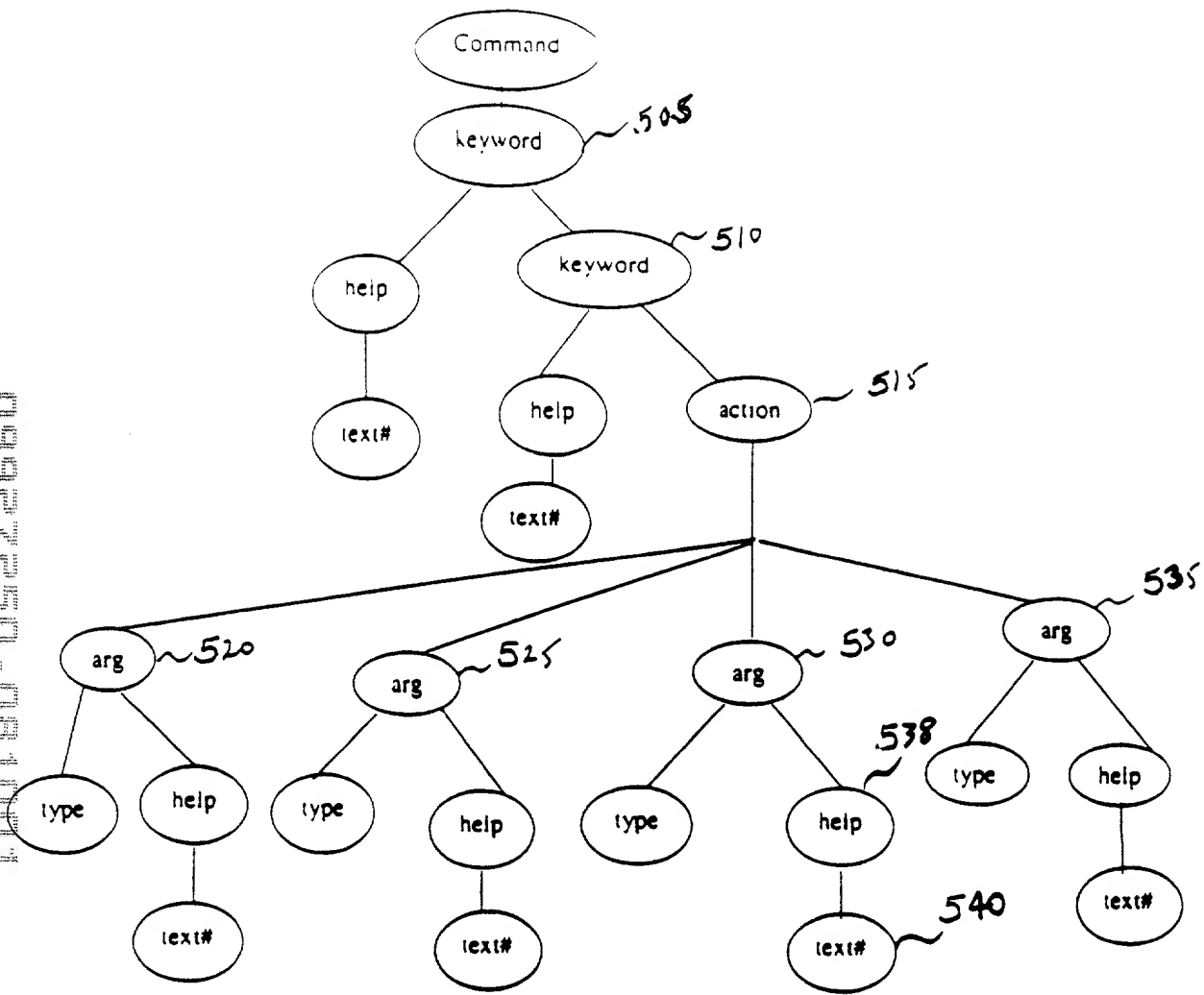


FIGURE 5

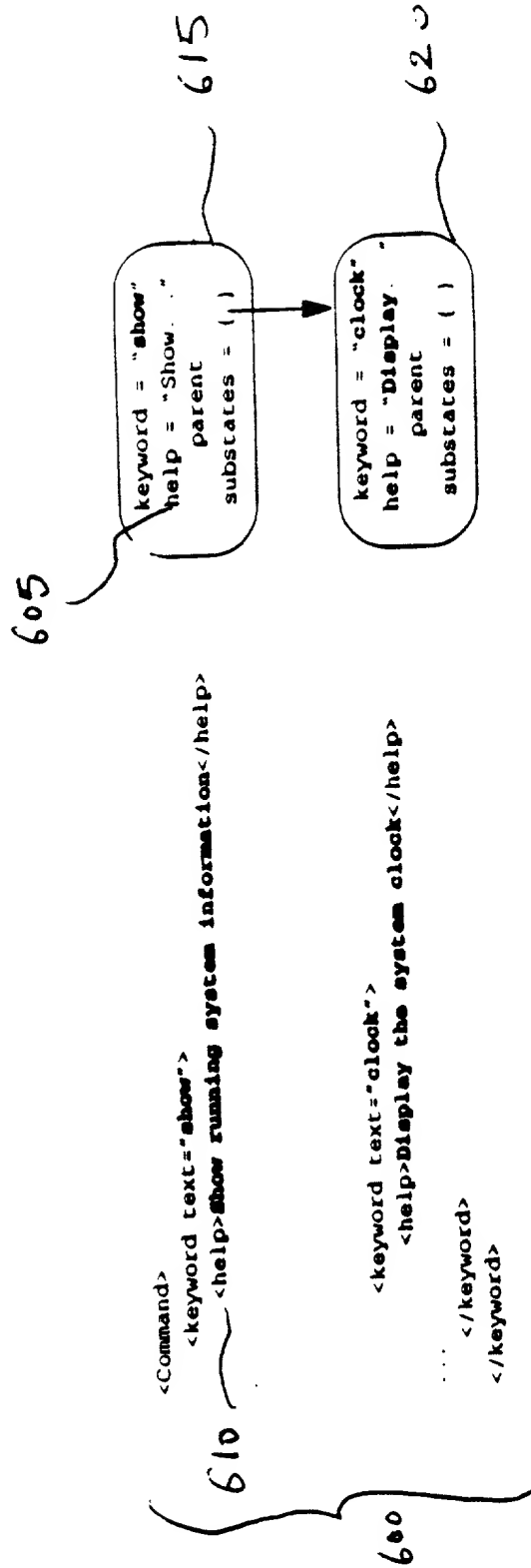


FIG. 6

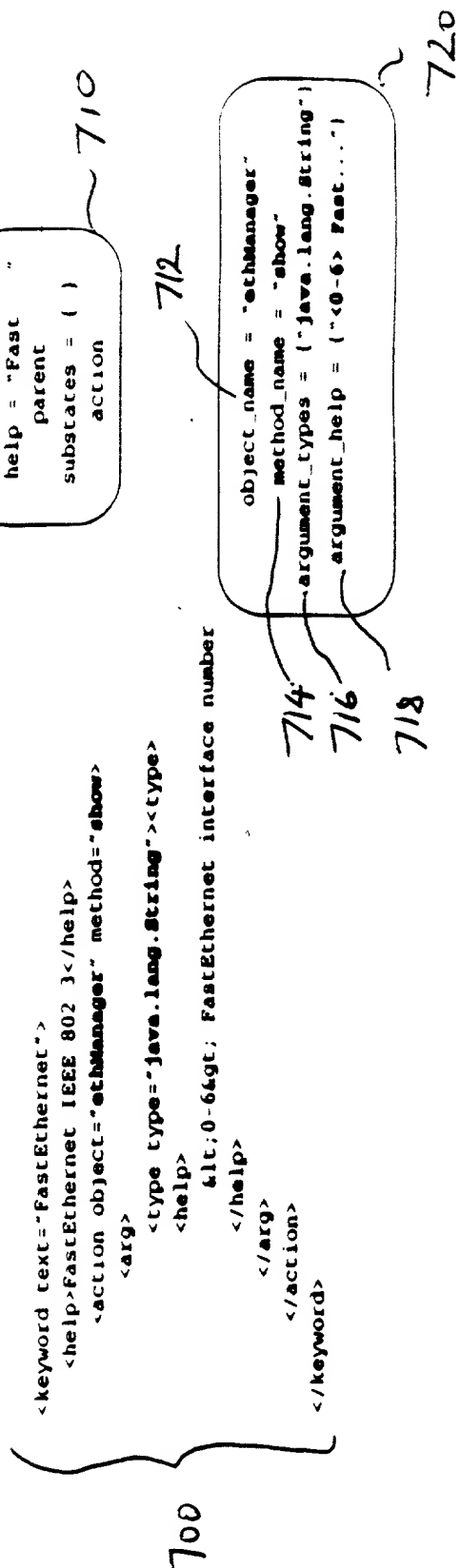
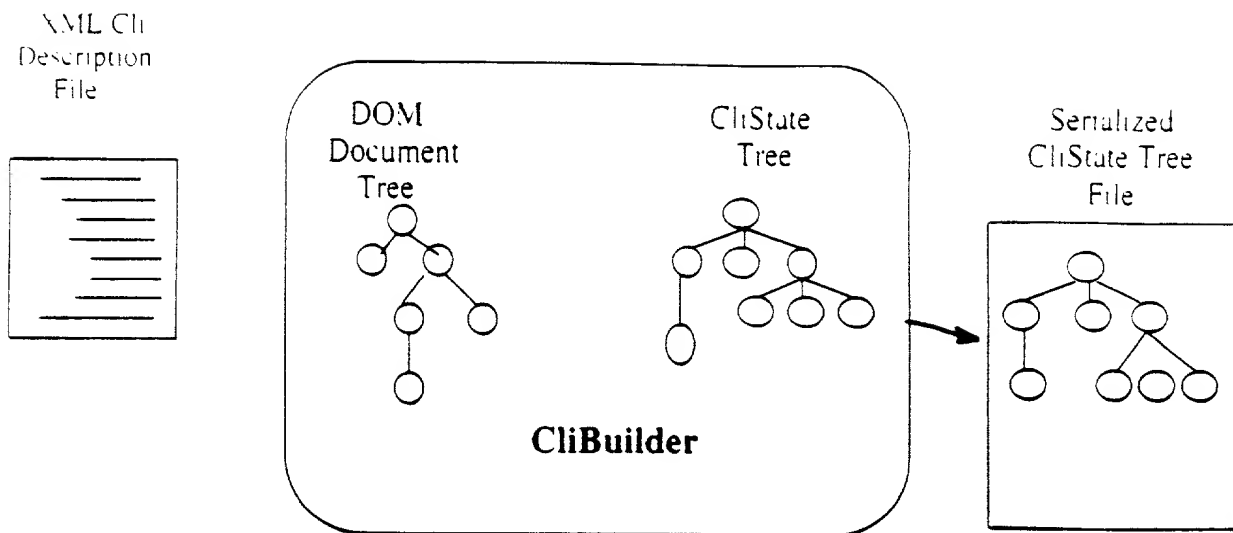
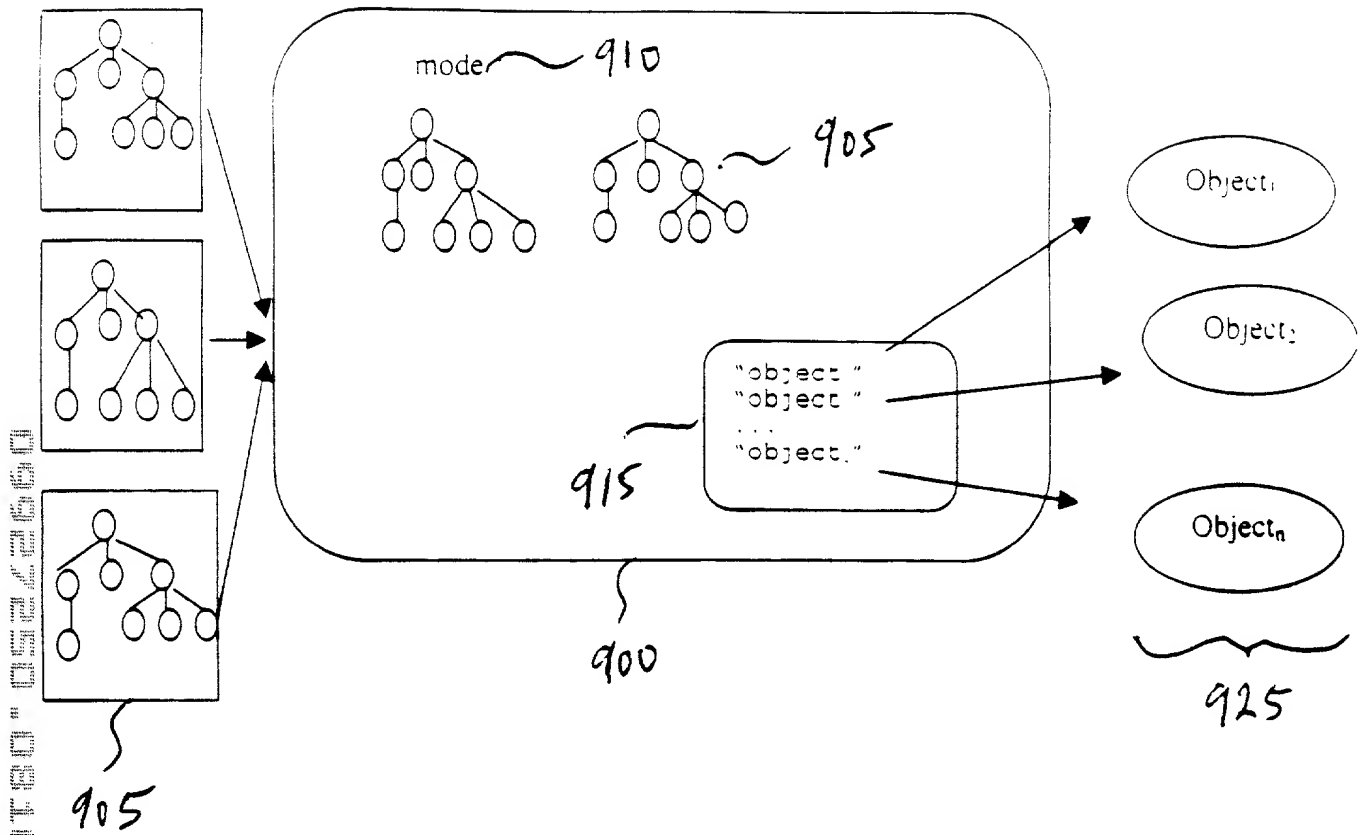


FIG. 7

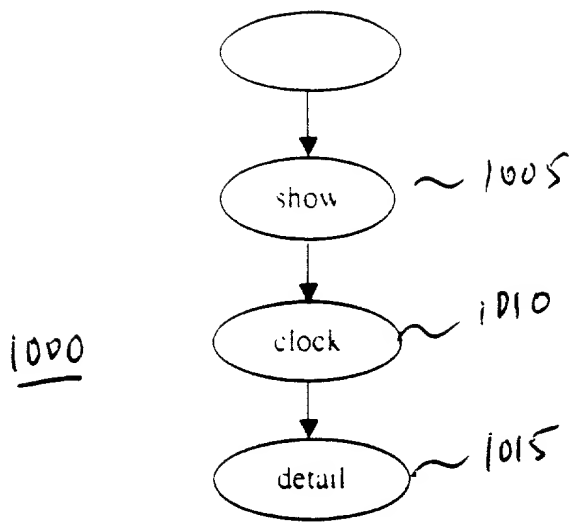


**FIGURE 8**

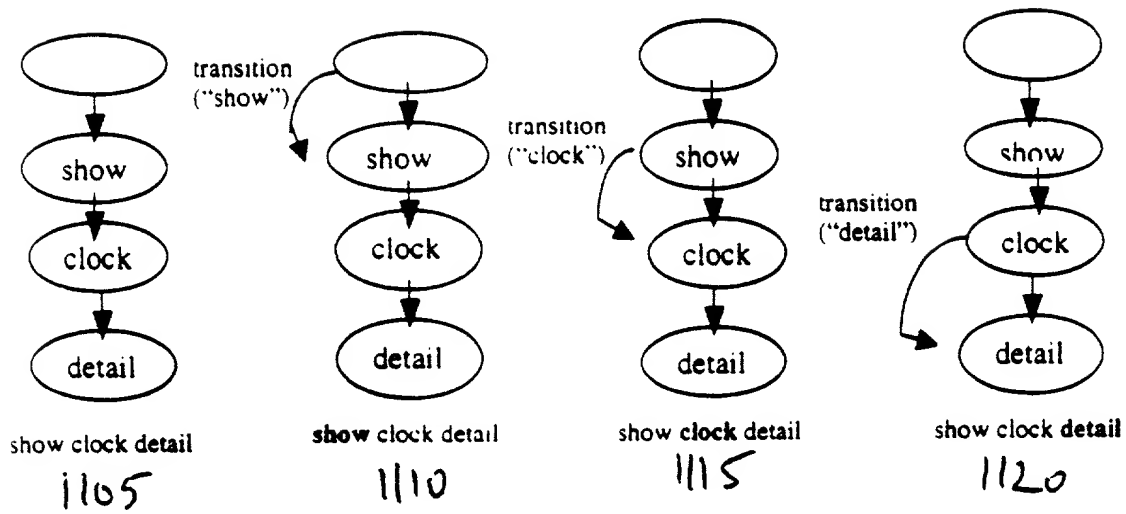




**FIGURE 9**

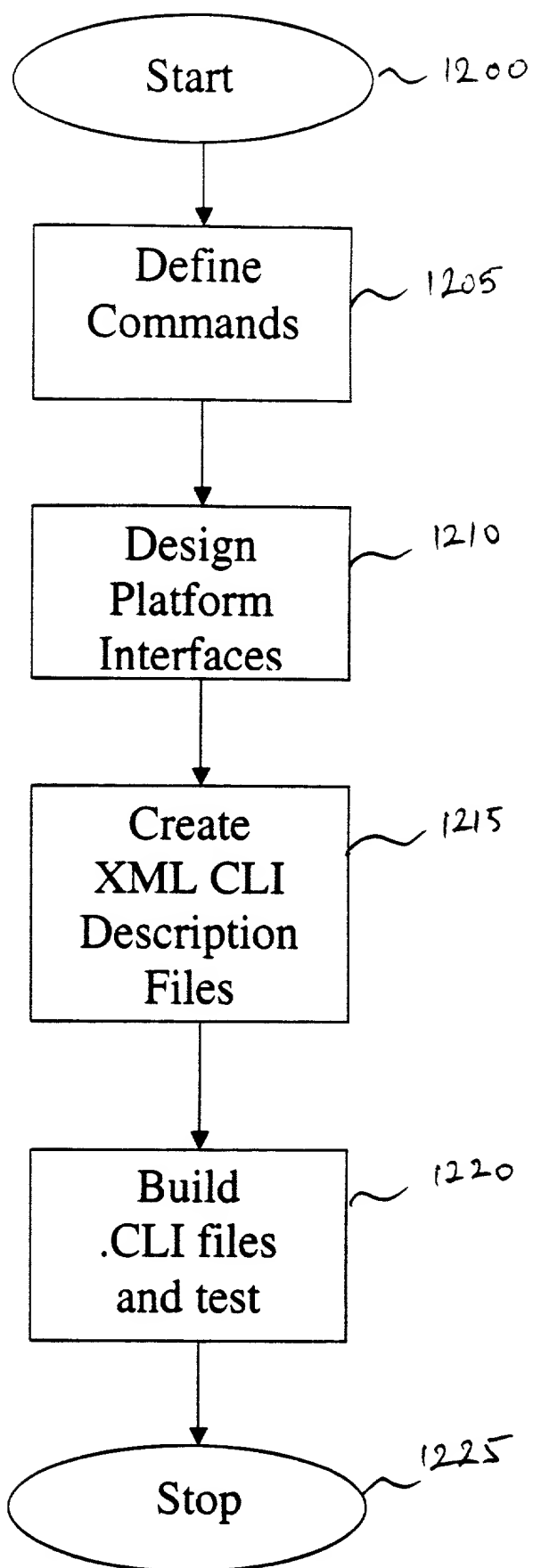


**FIGURE 10**



**FIGURE 11**

**FIGURE 12**



~ 1310

FIGURE 13

Argument Number	Argument Description	Argument Help
1	Time in 00 00:00 format	hh:mm:ss Current Time
2	day of the month in integer format	<1-31> Day of the month
3	month in three letter format	MONTH Month of the Year
4	Year in integer format	<1993-2035> Year

**FIGURE 14**

Command Keyword Sequence	Help
show	Show running system information
clock	Manage the system clock
show clock	Display the system clock
show clock detail	Display detailed information
clock set	Set the time and date

FIGURE 15

Argument Number	Argument Description	Java Type
1	Time in 00 00 00 format	java.lang.String
2	day of month integer format	java.lang.Integer
3	Month in three letter format	java.lang.String
4	Year in integer format	java.lang.Integer

**FIGURE 16**

```

<?xml version='1.0 encoding='US-ASCII' ?>
<! DOCTYPE cli SYSTEM "cli.dtd">
<cli>
  <mode name="exec">

    <command>
      <keyword text="show">
        <help>Show running system information</help>
        .... <!--lots of other show keywords... -->

          <keyword text="clock">
            </help>Display the system clock </help>
            <action object="Clock" method="show">
              </action>

            <keyword text="detail">
              <help>Display detailed information</help>
              action object="Clock" method="showDetail">
                </action>
              </keyword>
            </keyword>
          </command>

```

**FIGURE 17**

```

<command>
  <keyword text="clock"
  <help>Manage the System clock</help>

  <keyword test="set">
    <help> Set the time and date</help>
    <action object="Clock" method="set">

      <arg><!-- time argument -->
        <type type="Java.lang.String"></type>
        <help>hh:mm:ss Current time</help>
        </arg>

      <arg> <!-- day of the month argument -->
        <type type="Java.lang.Integer"></type>
        <help>&lt;1;1-31&gt; day of the month</help>
        </arg>

      <arg> <!-- day of the month argument -->
        <type type="Java.lang.Integer"></type>
        <help>MONTH Month of the year</help>
        </arg>

      <arg> <!-- year argument -->
        <type type="Java.lang.Integer"></type>
        <help>&lt;1;1993-2035&gt; Year</help>
        </arg>

    </action>
  </keyword>
</keyword>
</command>

```

**FIGURE 18**

```

public void show () {
    SimpleDateFormat formatter =
        new SimpleDateFormat ("hh:mm:ss.SSS zzz EEE
MMM dd yyyy");
    System.out.println("*" + formatter.format
(new Date()));
}

public void showDetail () {
    this.show();
    if (Clock.source())
        System.out.println ("Time source is user
configuration");
    else
        System.out.println ("No time source");
}

```

FIGURE 19



```

public void set (String time, Integer day, String month, Integer year)
    throws IOException, InterruptedException, CLIArgumentException
{
    // Validate the arguments
    SimpleDateFormat format = new SimpleDateFormat ("hh:mm:ss");
    format.setLenient (false);
    try { format.parse (time)
    } catch (ParseException e) {
        throw new ArgumentException (1);
    }

    if (day.intValue() < 1 || day.intValue() > 31)
        throw new ArgumentException (2);

    format.applyPattern ("MMM");
    try { format.parse(month);
    } catch (ParseException e) {
        throw new ArgumentException (3);
    }

    if (year.intValue() < 1993 || year.intValue() > 2035)
        throw new ArgumentException (4);

    // One last check for day, month and year validity (e.g., Feb 31)
    format.applyPattern ("dd MMM yyy");
    try { format.parse(day.toString() + month + year.toString());
    } catch (ParseException e) {
        CLIEngine.out.println("Invalid date (doesn't exist)");
        Return;
    }

    // Do the actual work
    String [] cmd = {"/bin/date", "-s", time + " " + day + " " + month + " " + year
        };
    Process proc = Runtime.getRuntime().exec (cmd);
    proc.waitFor();
    if (proc.exitValue() > 0) {
        CLIEngine.err.println("Operation not permitted");
        return;
    }

    clock.sourceSet();
}

```

FIGURE 20